

## CLAIMS

1. A method of loading a class file into a virtual machine, said class file being  
5 associated with an object-oriented class, and said virtual machine operating in  
an object-oriented computing system, said method comprising:

determining whether one or more components of said class have been  
marked to be loaded into said virtual machine; and

loading said one or more components of said class into said virtual  
10 machine when said determining determines that said one or more components  
of said class have been marked to be loaded into said virtual machine.

2. A method as recited in claim 1, wherein said method further comprises:

not loading one or more components of said class when said  
15 determining determines that said one or more components of said class have  
not been marked to be loaded into said virtual machine.

3. A method as recited in claim 1, wherein said method further comprises:

marking one or more components of said class for loading into said  
20 virtual machine.

4. A method as recited in claim 3,

wherein said marking is done by defining an attribute associated with  
said object-oriented class, and

25 wherein said determining operates to determine whether one or more  
components of said class have been marked to be loaded into said virtual  
machine based on said attribute associated with said object-oriented class.

5. A method as recited in claim 3, wherein said method further comprises:

30 providing an attribute portion in said class file that describes said  
attribute associated with said object-oriented class.

6. A method as recited in claim 5, wherein said attribute portion in said class file is implemented as an attribute table.

7. A method as recited in claim 5, wherein said attribute table includes offsets  
5 of one or more components of the class file with respect to said class file.

8. A method as recited in claim 6, wherein said determining whether one or more components of said class have been marked to be loaded into said virtual machine comprises:

10       initiating a first sequential read of said class file; and  
          determining whether an attribute table has been found for said class file.

9. A method as recited in claim 8, wherein said method further comprises:

15       initiating a second sequential read of said class file;  
          determining whether a component has been encountered; and  
          determining whether a component has a corresponding entry in said attribute table of said class file when said component has been encountered.

20   10. A class file suitable for loading into a virtual machine, said class file being associated with an object-oriented class, and said virtual machine operating in an object-oriented computing system, said class file comprising:

          a load attribute portion that includes information about one or more components of said class which have been marked to be loaded into said  
25 virtual machine.

11. A class file as recited in claim 10, wherein said attribute portion includes an attribute table.

30   12. A class file as recited in claim 11, wherein said attribute table includes offsets of one or more components of said class file.

13. A computer readable media including computer readable code representing a class file suitable for loading into a virtual machine, said class file being associated with an object-oriented class, and said virtual machine operating in an object-oriented computing system,

5            wherein said computer readable code representing said class file comprises computer readable code representing a load attribute portion of said class file, and

             wherein said attribute portion represents information about one or more components of said class that have been marked to be loaded into said  
10       virtual machine.

14. A computer readable media as recited in claim 13, wherein said attribute portion represents a load attribute table.

15       15. A computer readable media as recited in claim 13, wherein said attribute table includes representation of offsets associated with one or more components of said class file.

16. A method of loading a class file into a virtual machine, said class file being  
20       associated with an object-oriented class, and said virtual machine operating in an object-oriented computing system, said method comprising:

             providing a load attribute for said class file;

             associating one or more components of said class file with said load attribute to indicate that said one or more components of said class file are to  
25       be loaded; and

             loading only said one or more components of said class file into said virtual machine.

17. A method as recited in claim 16, wherein said providing of said load  
30       attribute operates to provide an attribute table in said class file.

18. A method as recited in claim 17, wherein said attribute table includes offsets of one or more components of the class file with respect to said class file.

5

19. A method as recited in claim 18, wherein said method further comprises:  
determining whether at least one component of said class file has been associated with said load attribute.

10 20. A method as recited in claim 18, wherein said determining operates to search said attribute table for an offset associated with said at least one component of said class file.

21. A method as recited in claim 16, wherein said determining comprises:  
15 initiating a first sequential read of said class file to determine whether said class file has an attribute table;  
reading said attribute table when said class file has an attribute table;  
initiating a second read of said class file; and  
determining whether at least one component of said class file has been  
20 associated with said load attribution.